Amendments to the Claims:

This listing of claims will replace all prior version, and listings, of claims in the application. Where claims have been amended and/or canceled, such amendments and/or cancellations are done without prejudice and/or waiver and/or disclaimer to the claimed and/or disclosed subject matter, and the applicant and/or assignee reserves the right to claim this subject matter and/or other disclosed subject matter in a continuing application.

Listing of Claims:

1. (Currently amended): A method of executing an instruction comprising:

determining if at least a portion of the instruction is stored in a loop buffer; and

determining if at least a portion of the instruction is stored in a cache;

wherein determining if at least a portion of the instruction is in a loop buffer includes determining if at least a portion of the instruction is in first portion of a memory array, and wherein determining if at least a portion of the instruction is in the cache includes determining if at least a portion of the instruction is in a second portion of the memory array.

2. (Original): The method of claim 1, wherein determining if at least a portion of the instruction is in a loop buffer includes determining if at least a portion of an address corresponding to the instruction is substantially equal to a tag address.

3. (Original): The method of claim 1, wherein determining if at least a potion of the instruction is in a loop buffer includes comparing at least a potion of an address corresponding to the instruction with at least a portion of a logic value stored in a tag register.

Claim 4 (Canceled)

5. (Currently amended): The method of claim [[4]] 1, wherein determining if at least a portion of the instruction is in a loop buffer includes determining if at least a portion of the

instruction is in first portion of a memory array, and determining if at least a portion of the instruction is in the cache includes determining if at least a portion of the instruction is in a second portion of the memory array, the first portion of the memory array being substantially contiguous with the second portion of the memory array.

6. (Original): The method of claim 1, further comprising loading a tag register after determining at least a portion of the instruction is in the cache.

7. (Original): The method claim 6, wherein loading the tag register includes loading a logic value that corresponds at least in part to a storage location in a memory array.

8. (Original): The method of claim 7, further comprising determining if at least a portion of an additional instruction is in a loop buffer by determining if at least a portion of the additional instruction corresponds to the logic value in the tag register.

9. (Currently amended): A method of processing data, wherein a memory array includes a loop buffer, comprising:

determining if a first piece of data is in the loop buffer; and

enabling a portion of the memory array corresponding to the loop buffer;

wherein the memory array further includes a cache, further comprising:

determining if a second piece of data is in the cache if the second piece of data is not in the loop buffer..

10. (Original): The method of claim 9, wherein enabling a portion of the memory array includes enabling only the portion of the memory array comprising the first piece of data.

Claim 11 (Canceled)

12. (Currently amended): The method of claim [[11]] 9, further comprising:

enabling the memory array if the second piece of data is not in the loop buffer.

13. (Currently amended): The method of claim [[11]] 9, further comprising:

loading a tag register with a first logic value corresponding, at least in part, to a location of the second piece of data in the memory array.

14. (Original): The method of claim 13, further comprising:

determining if a third piece of data is stored in the loop buffer by determining if the first logic value stored in the tag register corresponds, at least in part, to the third piece of data.

15. (Original): The method of claim 14, further comprising:

determining if the third piece of data is in the memory array if the third piece of data is not in the loop buffer; and

loading the tag register with a second logic value corresponding, at least in part, to a location of the third piece of data in the memory array.

16. (Original): The method of claim 15, wherein loading the tag register with the second logic value includes loading the tag register with a logic value that is different than the first logic value.

17. (Currently amended): The method of claim [[11]] 9, wherein determining if the second piece of data is in the cache includes determining if the second piece of data is in the memory array.

18. (Original): The method of claim 9, further comprising:

disabling a tag look-up of the memory array.

19. (Original) The method of claim 9, further comprising:

providing at least a portion of the first piece of data to a digital signal processing core.

Claims 20-23 (Canceled)